Solving an Optimal Control Problem of Cancer Treatment by Artificial Neural Networks

F. Heydarpour[1], E. Abbasi[2], M. J. Ebadi[3], S. M. Karbassi[1] *

[1] Department of Applied Mathematics, Yazd University, Yazd (Iran)
[2] Department of Computer Science, Yazd University, Yazd (Iran)
[3] Department of Mathematics, Chabahar Maritime University, Chabahar (Iran)

Abstract

Cancer is an uncontrollable growth of abnormal cells in any tissue of the body. Many researchers have focused on machine learning and artificial intelligence (AI) based on approaches for cancer treatment. Dissimilar to traditional methods, these approaches are efficient and are able to find the optimal solutions of cancer chemotherapy problems. In this paper, a system of ordinary differential equations (ODEs) with the state variables of immune cells, tumor cells, healthy cells and drug concentration is proposed to anticipate the tumor growth and to show their interactions in the body. Then, an artificial neural network (ANN) is applied to solve the ODEs system through minimizing the error function and modifying the parameters consisting of weights and biases. The mean square errors (MSEs) between the analytical and ANN results corresponding to four state variables are 1.54e-06, 6.43e-07, 6.61e-06, and 3.99e-07, respectively. These results show the good performance and efficiency of the proposed method. Moreover, the optimal dose of chemotherapy drug and the amount of drug needed to continue the treatment process are achieved.

Keywords

I. Introduction

CANCER is a class of diseases caused by disorders in the natural function of the body cells and it is one of the main reasons of death in the world. Development of cancer happens when the normal control mechanism of the body does not work. In this situation, the former cells do not die but they will grow without control and new abnormal cells are developed. This making additional cells may cause a mass of tissue which is called a tumor. There are many types of cancer, such as lung cancer, prostate cancer, breast cancer, and colorectal cancer. The main and usual treatments for cancer therapy are surgical procedures, hormone therapy, radiation therapy, organic therapy and chemotherapy.

The history of anticancer chemotherapy mathematical modeling comes back to more than forty years ago. Recently, plenty of attempts have been made to mathematically model the dynamics of the population of tumor cell and optimal control theory in order to provide optimal strategies for cancer treatment. New progress in computer and biological sciences have drawn attention to the role of computational methods in the wide scope of cancer research.

We are witnessing the improvement and development of many mathematical models for anticipating the growth of tumor after execution of chemotherapy and for inhibiting the disease progression

within treatment time. This can be done by minimizing the number of cancer cells, minimizing the doses of the drug as well as lessening side effects by the use of optimization methods in which finding optimal drug administration is considered as an optimal control problem (OCP) of a dynamical system [1]-[12]. The drug effect as well as interplays between normal, immune, and tumor cells in these models are seen as a result of the chemotherapy treatment drug for cancer. Based on the optimal measured dosage of the drug and the population number of tumor cells, Calzada et al. [9] and De Pillis and Radunskaya [6] proposed a performance index to decrease the tumor density. El-Gohary [7] investigated the optimal control of tumor, the disorders behavior and the stability of the system in the equilibrium states of tumor's model prior and posterior to the injection. Shuo Wang and Heinz Schattler [11] offered an optimal cancer therapy to decrease the tumor density and its destructive impacts during a specific time period. Urszula Ledzewicz et al. [12] argued the influences of tumor growth disorders and drug resistance on the process of treatment and optimal control analysis of a mathematical model.

The interplay between cancerous, immune, and healthy cells can be modeled as an ordinary differential equations (ODE) system [13]. To explain natural phenomena by using the differential equation, as an application of differential equations, one can convert the natural phenomena and related problems into the differential equations and then solve those differential equations. Since achieving the analytical solution of ODE's systems may not be easy, the researchers have attempted to find the approximate solutions for these systems. Many algorithms, for instance, finite element [14], predictor–corrector [15], finite difference, Runge–Kutta [16], and other methods have been studied and developed to solve these kinds of systems. In

* Corresponding author.

E-mail addresses: smkarbassi@yazd.ac.ir (S. M. Karbassi), dr.f.heidarpoor@gmail.com (F. Heydarpour), e.abbasi@yazd.ac.ir (E. Abbasi), ebadi@cmu.ac.ir (M. J. Ebadi).

these methods, the domain needs to be discretized into some finite domains or some points where we have the local approximation of the functions. Therefore, several AI methods especially artificial neural network (ANN) methods have been extensively applied to solve ODEs. Some of the related methods [17]-[30] are given in Table I.

TABLE I. Some ANN Methods for Solving ODEs

Reference	Method
[17]	Hopfield NN
[18][19]	Feedforward NN
[20]	Multi-layer perceptron (MLP)
[21] [22]	Hybrid of NN and optimization techniques
[23]	An unsupervised version of kernel least mean square algorithm
[24]	New algorithms based on NN
[27]	Multilayer perceptron and RBFNN
[29]	Multi-Quadric RBFNN
[30]	Regression-based NN
[41]	Adaptive resonance theory and boosted fuzzy classifier
[42]	ART1 network
[43] [44]	Machine Learning
[45]	Single layer Legendre Neural Network (LeNN)
[26] [46]	ANN
[47]	ANN time-series

In the field of computer sciences, the ANN has been known as a technology which has broad applications in many areas of engineering and science such as robot control, medicine, optimal control, manufacturing system design, pattern recognition, signal processing, business, classification and education (see [31]-[39] and the references therein). The real-time operation, Adaptive learning, Fault-tolerance through surplus information coding and self-organization are some benefits of ANNs.

An ANN is a configuration of artificial intelligence on the base of algorithms which imitate the human brain operation. It uses the existing patterns in the data and makes a generalization from these patterns to classify or predict. Neural networks (NNs) are effective particularly for interpreting the nonlinear data that usually occur in the domain of medicine and biology research. Fundamentally, it is a mathematical model inspired by the structure of systems of biological neurons, similar to the brain processes of the human. A neuron in the ANN is a computational unit whose inputs and outputs are numbers. Fig. 1 shows a simple neuron. Neuron inputs (X_1, X_2, X_3) have their own weights (W_1, W_2, W_3). The output of each neuron is calculated as follows:

$$output= f(W1X1+W2X2+W3X3)$$

The function f can be a binary step, linear or non-linear function such as sigmoid.
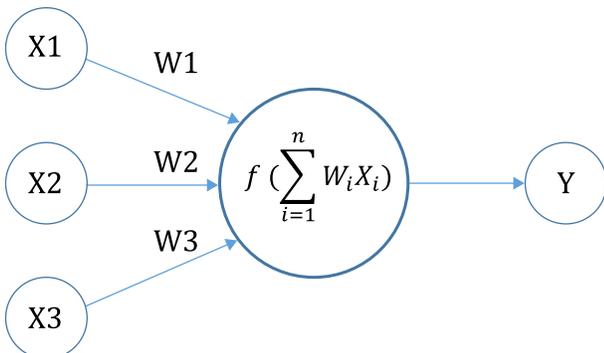

Fig. 1. A simple neuron (neuron is a computational unit, which its inputs and outputs are numbers).
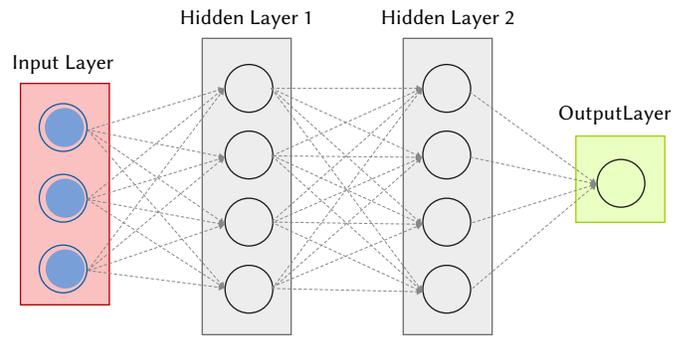

Fig. 2. An ANN Model (a feedforward network).

Fig. 2 is a sample fully connected feedforward neural network consisting of an input layer, two hidden layers and an output layer. The connections among these neurons similar to biological neurons specify the network's behavior which can be realized through the process of backpropagation [40]. In backpropagation, a given data set forms the network input, then network output is constantly compared with ANN output in order to minimize the MSE by making adjustments to the weights of the network. The network achieves higher accuracy after several repeated adjustments and becomes ready for solving complex problems [40]. Nowadays, it is found that biomedical systems should be demonstrated through nonlinear systems that represent ANNs as rich computing sources of biological investigation. In the last three decades, ANNs have been implemented in different domains of cancer medicine (several samples are given in Table I. [41]-[44]). To find the solutions of a nonlinear differential equation numerically, radial basis function (RBF) and multilayer perceptron neural networks with a novel unsupervised training model are utilized by Shirvany et al. [27]. Yazdi et al. [23] proposed and developed a new method based on a version of unsupervised kernel least mean square algorithm (KLMS) to solve ordinary differential equations of first and second order. A neural network on the base of regression was proposed by Mall and Chakraverty [30] to solve ordinary differential equations of higher and lower orders. Mall and Chakraverty [45] proposed a novel model for solving initial as well as boundary value problems on the base of Legendre Neural Network (LeNN) method with a single layer. A neural network model is proposed by Naushad et al. [46] for predicting the possibility of a woman growing breast cancer by analyzing the interplay of nutrients, genes, and demographic indicators. Another study having applications in quality warranty research and having modeled the treatment data resulted from the use of ANNs for cancer patients' radiotherapy demonstrated computation superiority for time-series ANN modeling over traditional techniques [47].

Although all the presented ordinary differential equations (ODE) systems used for cancer therapy have attempted to find the optimal doses of chemotherapy drugs, none of them has focused on the prediction of the drug dose needed to continue the treatment process. To fill the existing gap, the present work seeks new solutions to the system of ordinary differential equations (ODEs) for achieving the optimal dose of chemotherapy drugs and anticipating the amount of drug needed to continue the treatment process. To this end, a new method including Feedforward Neural Network (FFNN) is presented to find solutions to ODE at the same time predict cancer issues. Showing cancerous, immune and healthy cells interaction in the body and anticipating the tumor growth, this method predicts the optimum drug dose that should be injected with the most beneficial effect on tumor, immune, and normal cells. The dynamic behaviour of the system is analyzed by investigating the different equilibrium points in which the population number of tumor cells is zero or few. On the base of the ANN, the trial solution involves two components:

the first component with no adjustable parameters satisfies the initial/boundary conditions and the second component comprises the output of ANN. There are some advantages to the approximate solution of ANN comparing to the traditional numerical methods. First, we have the differentiability and closeness to the analytical solution for the approximate solution. Second, when the training points number increases, we do not encounter with more computation complexity. Third, the speed with which the parameters of the ANN (biases and weights) are modified increases to reduce error function. The proposed method is compared with two machine-learning methods, Support Vector Regression (SVR) and Multi-Layer Perceptron (MLP) as well as two well-known numerical methods, Finite Element and Runge–Kutta.

The organization of this paper is as follows. In Section II, the modeling of the tumor and its equilibrium points are presented. Section III gives the proposed method including the formulation of the first order ODE and the structure of ANN. The numerical results are discussed in Section IV. Finally, some conclusions are given in Section V.

II. The Tumor Model

The tumor expansion has attracted the attention of technicians, mathematicians and medical specialists, which can be modeled as systems of partial differential equations (PDEs) or ordinary differential equations (ODEs). The treatment of malignancy tumor is crucial. Hence, our model represents an optimal medical therapy for malignancy tumor derived from [13], [48]. The emphasis of this model is not on a specific kind of tumors, but it is a dynamic model to deal with the tumor expansion problem in its society [13], [48]. The explanation of our proposed model, its reduction and stability are brought in the following.

A. Tumor With Drug Mathematical Model and Stability of the Equilibrium States

The tissue close to the tumor site is targeted by the model consisting of three anomalous cell populations. $I(t)$, $T(t)$, and $N(t)$ illustrate immune cells, tumor cells, and normal cells at the time respectively. To show the impact of drug therapy, a state D is added representing the concentration of the drug in the bloodstream. The nonlinear differential system of equations expresses each component t dynamics for the treatment of chemotherapy which is shown below [13], [48].

$$\dot{T}(t) = T(t)[r_1(1 - b_1 T(t)) - c_2 I(t) - c_3 N(t) - a_1(1 - e^{-D(t)})];$$

$$\dot{N}(t) = N(t)[r_2(1 - b_2 N(t)) - c_4 T(t) - a_2(1 - e^{-D(t)})];$$

$$\dot{I}(t) = s + I(t)\left[\frac{r_3 T(t)}{(a + T(t))} - c_1 T(t) - d_1 - a_3(1 - e^{-D(t)})\right];$$

$$\dot{D}(t) = -d_2 D + \upsilon(t), \qquad (1)$$

where $\upsilon(t)$ is used for the amount of drug consumed orally or by injections before impacting on the bloodstream and d_2 is the number of death per capita for the drug. Here, because of physiological reasons, $I(t)$, $T(t)$ and $N(t)$ values and their parameters should be positive. A situation that a system does not alter is called the equilibrium point. When a dynamical system is represented by the differential equation to find the equilibrium point, one can take the first derivative of the system equal to zero.

Definition 2.1. A point $x^* \in R^n$ is called an equilibrium point of the ODEs system $\dot{x} = F(x)$ if it satisfies $F(x^*) = 0$, where

$$F(x) = \begin{bmatrix} F_1(x_1, x_2, \dots, x_n) \\ F_2(x_1, x_2, \dots, x_n) \\ \vdots \\ F_n(x_1, x_2, \dots, x_n) \end{bmatrix} [49].$$

In this case, a patient is called the treated patient if the dynamic system of tumor model without drug is reached to the equilibrium points. In the equilibrium points, the population number of tumor cells is zero or few. Therefore, it is necessary to find these points. After finding the equilibrium points of the system, it is essential to investigate the stability of the model [13], [49]. According to the parameters values of the system, there are three kinds of equilibria for a system without drug input as follows [13].

B. Tumor-free Equilibrium Point

In this case, the population of tumor cells is zero and the normal cells remain alive. If

$$r_1 < c_3 + {c_2 s}/{d_1}$$

then, we have the asymptotical stability of this equilibrium point whose form is as follows:

$$T_f^* = ({1}/{b_2}, 0, {s}/{d_1})$$

1. Dead Equilibrium Points

If the population number of normal cells is zero, this equilibrium point is said to be "dead". We have two types of these equilibrium points. In the first type, both population number of the tumor and normal cells are zero and the dead equilibrium point is as follows

$$D_1^* = (0, 0, {s}/{d_1})$$

in the second type, the population number of normal cells is zero but the tumor cells remain alive and the dead equilibrium point is as follows

$$D_2^* = (0, x, f(x))$$

Where x is a solution of (2) which is nonnegative value and $f(x)$ is introduced in (3):

$$x + \left(\frac{c_2}{r_1 b_1}\right) f(x) - \frac{1}{b_1} = 0 \qquad (2)$$

$$f(x) = \frac{s(x+a)}{c_1 x(x+a) + d_1(x+a) - \rho x} \qquad (3)$$

D_1^* is always an unstable equilibrium point. Depending on the system parameters values, we may have D_2^* as stable or unstable equilibrium point.

2. Coexisting Equilibrium Point

In this kind, both tumor and normal cells remain alive while they have coexistence and their population number is not zero. The form of this kind can be seen below:

$$C^* = (g(y), y, f(y))$$

where y is a solution of the following equation with nonnegative value:

$$y + \left(\frac{c_2}{r_1 b_1}\right) f(y) + \left(\frac{c_3}{r_1 b_1}\right) g(y) - \frac{1}{b_1} = 0 \qquad (4)$$

And $g(y)$ is as follows:

$$g(y) = {1}/{b_2} - ({c_4}/{r_2})y$$

Relying on the parameters of the system, this equilibrium point is fairly stable or unstable. For the chemotherapy remedy, the system must ideally come close to two equilibrium points including T_f^* and any coexisting equilibrium point when the value of y is small and

$g(y)$ is near to 1. The objective of chemotherapy treatment is that the system is brought into an attraction area in which equilibrium point can be tumor free or can be in the presence of only small population number of tumors.

C. Reducing Parameters of the System

In this section, we will decrease the parameters of the system by determining the new parameters of the system and defining new variables. In order to decrease the parameters of the system, we use some new variables as

$$\psi_1(t) = b_1 T(t), \psi_2(t) = b_2 N(t), \psi_3(t) = \frac{d_2 I(t)}{s}, \psi_4(t) = D(t),\ v(t) = d_2 u_4(t),\ \tau = d_2 t \tag{5}$$

We have the following system by substitution (5) into (1)

$$\dot{\psi}_1 = \psi_1\big[k_1(1-\psi_1) - n_2\psi_3 - n_3\psi_2 - m_1(1 - e^{-\psi_4})\big]$$
$$\dot{\psi}_2 = \psi_2\big[\ k_2(1-\psi_2) - n_4\psi_1 - m_2(1 - e^{-\psi_4})\big]$$
$$\dot{\psi}_3 = 1 + \psi_3\big[\frac{k_3\psi_1}{v_1 + \psi_1} - n_1\psi_1 - v_2 - m_3(1 - e^{-\psi_4})\big]$$
$$\dot{\psi}_4 = u - \psi_4 \tag{6}$$

Where the relations between the new parameters of the system and old parameters of the system are as follows:

$$k_i = {}^{r_i}/_{d_2},\qquad m_i = {}^{a_i}/_{d_2},\qquad (i = 1,2,3),$$
$$n_1 = {}^{c_1}/_{b_1 d_2},\quad n_2 = {}^{c_2 s}/_{d_2^{\ 2}},\ n_3 = {}^{c_3}/_{b_2 d_2},$$
$$n_4 = {}^{c_4}/_{b_1 d_2},\quad v_1 = \alpha b_1,\quad v_2 = {}^{d_1}/_{d_2} \tag{7}$$

In comparison with the system (1), solving the system (6) is simpler because of the reduction in the numbers of the system parameters.

III. Proposed Method

A. First Order ODEs System Formulation

Now, a system of first order initial value differential equations is considered as follows:

$$\frac{d\psi_k}{dt} = f_k\big(t, \psi_1, \psi_2, \psi_3, \psi_4\big), k = 1,2,3,4 \ \text{and}\ t \in\ [a,b] \tag{8}$$

Where $\psi^k(a) = A_k,\ k = 1,2,3,4$ and ψ_{T_k} indicates the kth trial solution based on ANN which can be shown here as:

$$\psi_{T_k}(t, p_k) = A_k + (t-a)N_k(t, p_k), k = 1,2,3,4 \tag{9}$$

Where $N_k(t, p_k)$ is the ANN's output corresponding to input data t and parameters p_k. The initial condition is satisfied by the trial solution $\psi_{T_k}(t, p_k)$. The derivative of the trial solution $\psi_{T_k}(t, p_k)$ is as follows:

$$\frac{d\psi_{T_k}(t,p_k)}{dt} = N_k(t, p_k) + (t - a)\frac{dN_k(t,p_k)}{dt}, k = 1,2,3,4 \tag{10}$$

B. The Error Function and Elements of the ANN

In this case, the error function is given with

$$E(p) = \sum_{i=1}^{n}\sum_{k=1}^{4}[\frac{d\psi_{\tau_k}(t_i,\ p_k)}{dt}$$
$$-f_k(t_i, \psi_{\tau_1}(t_i,p_1), \psi_{\tau_2}(t_i,p_2), \psi_{\tau_3}(t_i,p_3), \psi_{\tau_4}(t_i,p_4))]^2 \tag{11}$$

where n is the number of samples. The error function is composed of two terms. The first term described in (10) consists of the network output and its derivative. The second term involves the desired output. The error function (11) is minimized by applying *fminunc*

according to (12) which is a toolbox function of Matlab for solving the unconstrained optimization problem.

$$X = fminunc(fun, X_0) \tag{12}$$

In equation (12), the output of the *fminunc* is the local minimum of the function described in *fun* with respect to the parameters initialized in X_0. In this case, *fun* is the error function that is presented in (11) and the parameters are weights that are initialized randomly. With regard to a multilayer perceptron neural network with three layers including one input neuron and one output neuron in the input and output layers and a hidden layer with m neurons, the neural network output is computed for the input t as below

$$N(t, p) = \sum_{j=1}^{m} v_j\sigma(z_j) \tag{13}$$

where m is the number of neurons in the hidden layer, v_j's are the weights between hidden and output layers corresponding to j^{th} neuron in the hidden layer, and the activation function of each neuron in the hidden layer is considered as sigmoid given by:

$$\sigma(z_j) = \frac{1}{1+e^{-z_j}} \tag{14}$$

where z_j is:

$$z_j = w_j t + u_j \tag{15}$$

where w_j's are weights between input and hidden layers and the biases indicated by u_j. The network output derivative with respect to input t is as follows:

$$\frac{dN(t,p)}{dt} = \sum_{j=1}^{m} v_j w_j \sigma'(z_j) \tag{16}$$

In the next section, the numerical results will be shown while this strategy is applied for therapy and control of cancer modeled as an ODEs system.

IV. Numerical Results

In this section, the results of ANN are compared with the analytical solution using MATLAB R2015b, on a 2.4 GHz Intel(R) Core(TM) i7-4510U laptop running Windows 10 Ultimate with 12.00 GB main memory. The values of the ODEs system parameters presented in Section II are determined according to the following [7]:

$$k_1 = 30,\ k_2 = 48, k_3 = 29, n_1 = 2, n_2 = 1.3, n_3 = 0.47,$$
$$n_4 = 8, m_1 = 9, m_2 = 15, m_3 = 4, v_1 = 0.25, v_2 = 5,$$
$$\gamma_1 = 10, \gamma_2 = 20, \gamma_3 = 8, \gamma_4 = 5, \gamma = 15$$

By considering these parameter values, a four-coupled first order ODEs system is obtained as follows:

$$\dot{\psi}_1 = \psi_1\big[30(1-\psi_1) - 1.3\psi_3 - 0.47\psi_2 - 9(1 - e^{-\psi_4})\big]$$
$$\dot{\psi}_2 = \psi_2[48(1-\psi_2) - 8\psi_1 - 15(1 - e^{-\psi_4})]$$
$$\dot{\psi}_3 = 1 + \psi_3\big[\frac{29\psi_1}{0.25 + \psi_1} - 2\psi_1 - 5 - 4(1 - e^{-\psi_4})\big]$$
$$\dot{\psi}_4 = u - \psi_4, \tag{17}$$

where the initial conditions are $\psi_1(0) = 2.5,\ \psi_2(0) = 2.5, \psi_3(0) = 1.55$ and $\psi_1(0) = 1.35$. According to (9), the trial solutions are given by

$$\psi_{T1}(t, p) =\ 2.5 + tN_1(t, p_1),$$
$$\psi_{T2}(t, p) =\ 0.25 + tN_2(t, p_2),$$
$$\psi_{T3}(t, p) =\ 1.55 + tN_3(t, p_3),$$
$$\psi_{T4}(t, p) =\ 1.35 + tN_4(t, p_4).$$

According to the proposed method in Section III, four NNs

associated with ψ_1 to ψ_4 equations are trained with 156 samples by considering the step size of 0.0008 within the interval [0 0.125]. Fig. 3 illustrates the ANN and analytical results. The analytical results are obtained by Pontryagin minimum principle.

As it can be seen in Fig. 3, ANN results are a good approximation of analytical solutions because the MSEs between the analytical and ANN results corresponding to equations ψ_1 to ψ_4 are 1.54e-06, 6.43e-07, 6.61e-06 and 3.99e-07, which are denoted by E_1, E_2, E_3, and E_4, respectively.

In Table II, a comparison is provided between analytical solutions and ANN results for the different number of neurons in the interval [2, 8] with the step size 2. In Table II, for each equation except for ψ_4, the best results are obtained for six neurons in the hidden layer. As it can be seen from Table II, when the number of neurons in the hidden layer increases from 2 to 4, the MSEs decrease to the amount of 1.5206e-03, 2.8931e-03, and 4.709e-06, for ψ_1, ψ_3, and ψ_4 while the MSE does not change for equation ψ_2. In addition, when the number of neurons increases from 4 to 6, the MSEs of the equations ψ_1, ψ_2, and ψ_3, except for ψ_4, increase. By increasing the number of neurons from 6 to 8, the MSEs increase for ψ_1, ψ_2, and ψ_3, whereas the MSE decreases for ψ_4. The running time of ANN corresponding to each equation ψ_1 to ψ_4 is given in Table III.

TABLE II. Mean Square Error Between Analytical and ANN Results for 2, 4, 6 and 8 Neurons in Hidden Layer of ANN

Number of Neurons \ MSEs	2	4	6	8
ψ_1	0.0016	7.94e-05	1.54e-06	3.44e-06
ψ_2	0.00013	0.00013	6.43e-07	0.00013
ψ_3	0.0029	6.84e-06	6.61e-06	0.00013
ψ_4	5.07e-06	3.61e-07	3.99e-07	3.84e-07

TABLE III. Running Time of ANN in Seconds for Six Neurons in Hidden Layer

Optimal control equation	ψ_1	ψ_2	ψ_3	ψ_4
Running time	11.88	9.46	9.44	8.84

Ideally, the modeling should improve treatment strategies and lead to the development of new approaches. One of the goals of mathematical models is dosimetry, accurate prediction of dose response or simulation of different dose programs. Fig. 3 (d) shows the amount of drug used for different time intervals during treatment, and Figs. 3(a), 3(b) and 3(c) show the effect of the drug on tumor, normal and immune cells, respectively.

Fig. 3(d) shows that we increase the amount of drug used to counteract tumor cells till t = 0.08. However, by decreasing tumor cells and increasing immunity after t=0.08, we gradually decrease the amount of drug. From now on, tumor cells will decline due to the competition between them and immune cells. In other words, as the immune system increases, the amount of used drug and the tumor cells decrease. Where the amount of normal cells is increasing to prevent additional toxicity of chemotherapy injection, we try to keep them at 0.25 during treatment. To prevent self-control of the immune system on cancer cells, we assume the number of primary immune cells so small and the number of tumor cells so large that tumor growth is unavoidable unless chemotherapy is used. Support Vector Regression (SVR) is a version of SVM for solving regression problem which was proposed by Drucker, H. et al. [50].
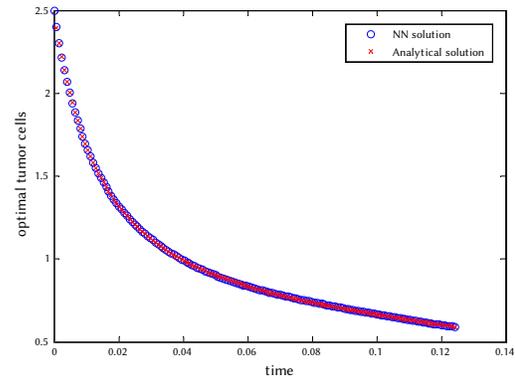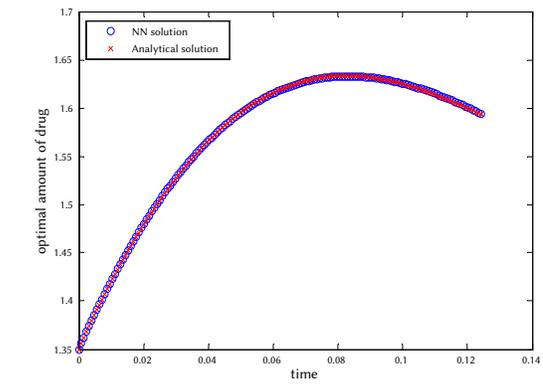

(a)
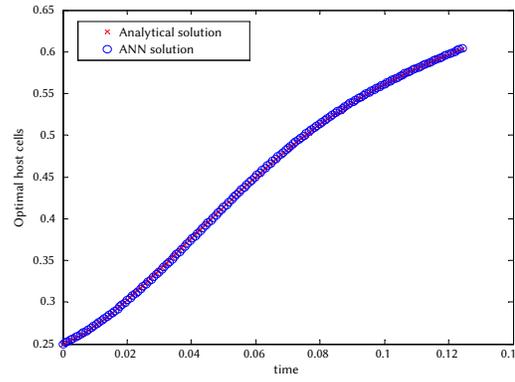
(b)
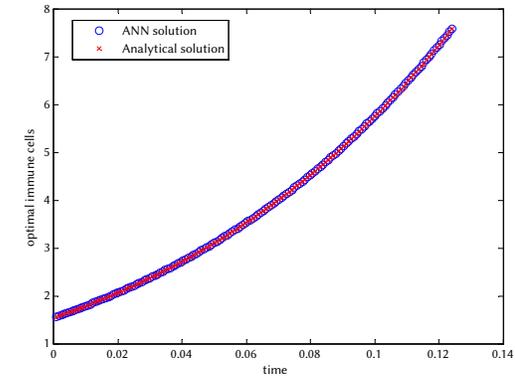
(c)

(d)

Fig. 3. Analytical and ANN solution for (a) Optimal tumor cells, (b) Optimal host cells, (c) Optimal immune cells and (d) Optimal amount of drug.

Currently, therapies are expensive and time-consuming. They often result in variable treatment outcomes among patients and their harmful effects are often unpredictable [51]. We compare the results of ANN with the SVR model. The Optimal amount of drug by using SVR and ANN is shown in Fig. 4. The MSEs between the analytical and SVR results corresponding to equations are 1.48, 0.007, 3.5816 and 0.0583 respectively. By comparing the results of ANN and SVR, it can be clearly seen that ANN performs better than SVR. In addition, Table IV shows the MSEs comparisons between the proposed method, SVR, MLP, Finite Element, and Runge–Kutta methods. We can clearly see that the proposed method provides more accurate results compared to the SVR and MLP machine learning methods as well as Finite Element and Runge–Kutta numerical methods.
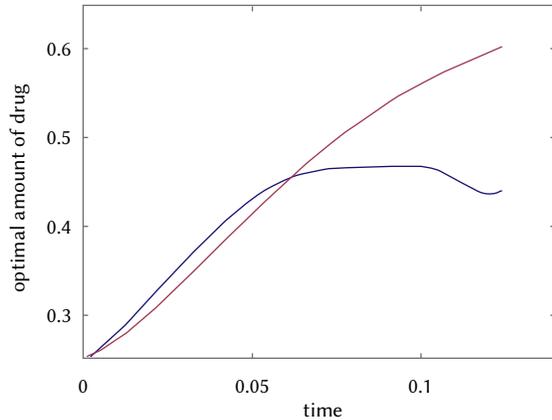

Fig. 4. Optimal amount of drug by using SVR(Red) and ANN(Blue).

Remark 4.1: In this paper, the goal is not just to find solutions to the ODEs system because the given ODE system can be solved by different numerical methods. Solving the system, the NN allows us to anticipate the process of cancer treatment here providing a great advantage over other methods for solving ODE systems. Using the proposed NN, the optimal dose of the chemotherapy drug injected is shown in Fig. 3 (d).

TABLE IV. MSEs Comparisons Between SVR, MLP, Finite Element, Runge–Kutta, and Proposed Method

| MSEs \ Methods	SVR	Finite Element	Here	MLP	Runge-Kutta
ψ_1	1.48	3.24e-04	**1.54e-06**	7.29e-05	1.003
ψ_2	0.007	2.12e-05	**6.43e-07**	8.78e-06	0.002
ψ_3	3.5816	1.37e-04	**6.61e-06**	5.61e-05	1.247
ψ_4	0.0583	7.51e-05	**3.99e-07**	7.82e-07	0.0018

V. Conclusion

In this paper, a system of ODEs was used to anticipate the tumor growth and to show cancerous, immune and healthy cells interaction in the body. Also, the dynamic behaviour of the system was analyzed by investigating the different equilibrium points in which the population number of tumor cells is zero or few. Using the proposed ANN, not only the optimal dose of chemotherapy drug that should be injected was shown, but also the amount of the drug needed to continue the treatment process was predicted. This can lead to decreasing the number of tumor cells and increasing the number of immune and healthy cells to their utmost. Simulation results had shown the good performance and efficiency of the proposed method. Compared to existing methods in cancer treatment and diagnostic centers that require advanced equipment with expensive costs, this method can be used without trial and error to predict the process of cancer treatment. In the future works, we will design and implement effective closed loop control systems to control the dose of a chemotherapy drug to reduce the side effects of receiving an additional drug.

References

[1] F. Michor and K. Beal, "Improving cancer treatment via mathematical modeling: surmounting the challenges is worth the effort," *Cell 163*, vol. 3, no. 5, pp. 1059-1063, 2015, doi: 10.1016/j.cell.2015.11.002.

[2] P. Khalili, R. Vatankhah, and S. Taghvaei, "Optimal sliding mode control of drug delivery in cancerous tumor chemotherapy considering the obesity effects", *IET Systems Biology*, vol. 12, no. 4, pp. 185-189, 2018, doi: 10.1049/iet-syb.2017.0094.

[3] X. Yang, H. He, D. Liu, and Y. Zhu, "Adaptive dynamic programming for robust neural control of unknown continuous-time non-linear systems," *IET Control Theory & Applications*, vol. 11, no. 14, pp. 2307-2316, 2017, doi: 10.1049/iet-cta.2017.0154.

[4] B. Ribba, B. You, M. Tod, P. Girard, B. Tranchand, V. Trillet-Lenoir, and G. Freyer, "Chemotherapy may be delivered based on an integrated view of tumor dynamics," *IET Systems Biology*, vol. 3, pp.180-190, 2009, doi: 10.1049/iet-syb.2008.0104.

[5] E. Shakeri, G. Latif-Shabgahi, and A. E. Abharian, "Adaptive non-linear control for cancer therapy through a Fokker–Planck observer," *IET Systems Biology*, vol. 12, no. 2, pp. 73-82, 2018, doi: 10.1049/iet-syb.2017.0032.

[6] L. G. De Pillis and A. Radunskaya, "A mathematical tumor model with immune resistance and drug therapy: an optimal control approach," *Computational and Mathematical Methods in Medicine*, vol. 3, no. 5, pp. 79-100, 2001, doi: 10.1080/10273660108833067.

[7] A. El-Gohary, "Chaos and optimal control of equilibrium states of tumor system with drug," *Chaos Solitons & Fractals*, vol. 41, no. 2, pp. 425-435, 2009, doi: 10.1016/j.chaos.2008.02.003.

[8] M. Engelhart, S. Lebiedz, and S. Sager, "Optimal control for selected cancer chemotherapy ODE models: a view on the potential of optimal schedules and choice of objective function," *Mathematical Biosciences*, vol. 229, no. 1, pp. 123-134, 2011, doi: 10.1016/j.mbs.2010.11.007.

[9] M. C. Calzada, E. Fernández-Cara, and M. Marín, "Optimal control oriented to therapy for a free-boundary tumor growth model," *Journal of Theoretical Biology*, vol. 325, no. 7, pp. 1-11, 2013, doi: 10.1016/j.jtbi.2013.02.004.

[10] T. Agrawal, M. Saleem, and S. K. Sahu, "Optimal control of the dynamics of a tumor growth model with Hollings' type-II functional response," *Computational and Applied Mathematics*, vol. 33, no. 6, pp. 591–606, 2014, doi: 10.1007/s40314-013-0083-x.

[11] S. Wang and H. Schattler, "Optimal control of a mathematical model for cancer chemotherapy under tumor heterogeneity," *Mathematical Biosciences and Engineering*, vol. 13, pp. 1223-1240, 2016, doi: 10.3934/mbe.2016040.

[12] U. Ledzewicz, S. Wang, H. Schattler, N. André, M. A. Heng, and E. Pasquier, "On drug resistance and metronomic chemotherapy: A mathematical modeling and optimal control approach," *Mathematical Biosciences and Engineering*, vol. 14, pp. 217-235, 2017, doi: 10.3934/mbe.2017014.

[13] L. G. Pillis and A. Radunskaya, "The dynamics of an optimally controlled tumor model: A case study," *Mathematical and Computer Modelling*, vol. 37, no. 11, pp. 1221-1244, 2003, doi: 10.1016/S0895-7177(03)00133-X.

[14] J. N. Reddy, *An introduction to the finite element method*, New York, USA: McGraw—Hill, 2006.

[15] J. Douglas and B. F. Jones, "On predictor–corrector methods for nonlinear parabolic differential equations," *Journal of the Society for Industrial and Applied Mathematics*, vol. 11, no. 1, pp. 195–204, 1963, doi: 10.1137/0111015.

[16] A. Wambecq, "Rational Runge–Kutta methods for solving systems of ordinary differential equations," *Computing*, vol. 20, pp. 333-342, 1978, doi: 10.1007/BF02252381.

[17] H. Lee and I. S. Kang, "Neural algorithms for solving differential equations," *Journal of Computational Physics*, vol. 91, no. 1, pp. 110–117, 1990, doi: 10.1016/0021-9991(90)90007-N.

[18] A. J. Meade Jr and A. A. Fernandez, "The numerical solution of linear

ordinary differential equations by feed forward neural networks," *Mathematical and Computer Modelling,* vol. 19, no. 12, pp. 1-25, 1994, doi: 10.1016/0895-7177(94)90095-7.

[19] A. J. Meade Jr and A. A. Fernandez, "Solution of nonlinear ordinary differential equations by feed forward neural networks," *Mathematical and Computer Modelling,* vol. 20, no. 9, pp. 19-44, 1994, doi: 10.1016/0895-7177(94)00160-X.

[20] I. E. Lagaris, A. Likas, and D. I. Fotiadis, "Artificial neural networks for solving ordinary and partial differential equations," *IEEE Transactions on Neural Networks,* vol. 9, no. 5, pp. 987-1000, 1998, doi: 10.1109/72.712178.

[21] B. Liu and B. Jammes, "Solving ordinary differential equations by neural networks," in *13th European Simulation Multi-Conference Modelling and Simulation: A Tool for the Next Millennium,* Warsaw, Poland, June 14, 1999.

[22] A. Malek and R. S. Beidokhti, "Numerical solution for high order deferential equations using a hybrid neural network-optimization method," *Applied Mathematics and Computation,* vol. 183, no. 1, pp. 260-271, 2006, doi: 10.1016/j.amc.2006.05.068.

[23] H. S. Yazdi, M. Pakdaman, and H. Modaghegh, "Unsupervised kernel least mean square algorithm for solving ordinary differential equations," *Neurocomputing,* vol. 74, no. 12-13, pp. 2062-2071, 2011, doi: 10.1016/j.neucom.2010.12.026.

[24] J. Abdul Samath and N. Selvaraju, "Solution of matrix Riccati differential equation for nonlinear singular system using neural networks," *International Journal of Computer Applications,* vol. 1, no. 29, pp. 48–54, 2010, doi: 10.1.1.206.3895.

[25] L. P. Aarts and P. Van der Veer, "Neural network method for solving partial differential equations," *Neural Processing Letters,* vol. 14, pp. 261–271, 2001, doi: 10.1023/A:1012784129883.

[26] S. A. Hoda and H. A. Nagla, "Neural network methods for mixed boundary value problems," *International Journal of Nonlinear Science,* vol. 11, no. 3, pp. 312–316, 2011.

[27] Y. Shirvany, M. Hayati, and R. Moradian, "Multilayer perceptron neural networks with novel unsupervised training method for numerical solution of the partial differential equations," *Applied Soft Computing,* vol. 9, no. 1, pp. 20-29, 2009, doi: 10.1016/j.asoc.2008.02.003.

[28] K. McFall and R. Mahan, "Artificial neural network for solution of boundary value problems with exact satisfaction of arbitrary boundary conditions," *IEEE Transactions on Neural Networks,* vol. 20, no. 8, pp. 1221–1233, 2009, doi: 10.1109/TNN.2009.2020735.

[29] N. Mai-Duy and T. Tran-Cong, "Numerical solution of differential equations using multi quadric radial basis function networks," *Neural Networks,* vol. 14, no. 2, pp. 185-199, 2001, doi: 10.1016/S0893-6080(00)00095-2.

[30] S. Mall and S. Chakraverty, "Regression based neural network training for the solution of ordinary differential equations," *International Journal of Mathematical Modelling and Numerical Optimisation,* vol. 4, no. 2, pp. 136-149, 2013, doi: 10.1504/IJMMNO.2013.055203.

[31] R. T. Rockafellar, "Linear-quadratic programming and optimal control," *SIAM Journal on Control and Optimization,* vol. 25, no. 3, pp. 781-814, 1987, doi: 10.1137/0325045.

[32] Y. S. Xia, G. Feng, and J. Wang, "A primal-dual neural network for online resolving constrained kinematic redundancy in robot motion control," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics),* vol. 35, no. 1, pp. 54-64, 2005, doi: 10.1109/TSMCB.2004.839913.

[33] D. Bertsekas, *Constrained optimization and Lagrange multiplier methods,* New York, USA: Academic press, 2014.

[34] J. Wang, "Primal and dual neural networks for shortest-path routing," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans,* vol. 28, no. 6, pp. 864-869, 1998, doi: 10.1109/3468.725357.

[35] Q. Tao, X. Liu, and X. Cui, "A linear optimization neural network for associative memory," *Applied Mathematics and Computation,* vol. 171, no. 2, pp. 1119-1128, 2005, doi: 10.1016/j.amc.2005.01.123.

[36] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, *Nonlinear programming: theory and algorithms,* New York, USA: John Wiley & Sons, 2013.

[37] M. J. Ebadi, A. Hosseini, and M. M. Hosseini, "A projection type steepest descent neural network for solving a class of nonsmooth optimization problems," *Neurocomputing,* vol. 235, pp. 164-181, 2017, doi: 10.1016/j.neucom.2017.01.010.

[38] M. J. Ebadi, M. M. Hosseini, and S. M. Karbassi, "An efficient one-layer recurrent neural network for solving a class of nonsmooth pseudoconvex optimization problems," *Journal of Theoretical and Applied Information Technology,* vol. 96, no. 7, pp. 1999-2015, 2018.

[39] F. C. Morabito, M. Campolo, C. Ieracitano, J. M. Ebadi, L. Bonanno, A. Bramanti, S. Desalvo, N. Mammone, and P. Bramanti, "Deep convolutional neural networks for classification of mild cognitive impaired and Alzheimer's disease patients from scalp EEG recordings," in *IEEE 2nd International Forum on Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI),* Bologna, Italy, pp. 1-6, 2016, doi: 10.1109/RTSI.2016.7740576.

[40] P. J. Drew and J. R. T. Monson, "Artificial neural networks," *Surgery,* vol. 127, no. 1, pp. 3-11, 2000, doi: 10.1067/msy.2000.102173.

[41] H. Takahashi, Y. Murase, T. Kobayashi, and H. Honda, "New cancer diagnosis modeling using boosting and projective adaptive resonance theory with improved reliable index," *Biochemical Engineering Journal,* vol. 33, no. 2, pp. 100-109, 2007, doi: 10.1016/j.bej.2006.08.004.

[42] S. Swathi, G. Babu, R. Sendhilkumar, and S. N. Bhukya, Performance of ART 1 Network in the Detection of Breast Cancer, in *International conference on Computer design and Engineering (ICCDE),* Singapore 2015, pp. 100-105.

[43] K. Kourou, T. P. Exarchos, K. P. Exarchos, M. V. Karamouzis, D. I. Fotiadis, "Machine learning applications in cancer prognosis and prediction," *Computational and Structural Biotechnology Journal,* vol 13, pp. 8-17, 2015, doi: 10.1016/j.csbj.2014.11.005.

[44] J. A. Cruz and D. S. Wishart, "Applications of machine learning in cancer prediction and prognosis," *Cancer Informatics,* vol. 2, pp. 59-77, 2006, doi: 10.1177/117693510600200030.

[45] S. Mall and S. Chakraverty, "Application of Legendre Neural Network for solving ordinary differential equations," *Applied Soft Computing,* vol. 43, pp. 347-356, 2016, doi: 10.1016/j.asoc.2015.10.069.

[46] S. M. Naushad, M. J. Ramaiah, M. Pavithrakumari, J. Jayapriya, T. Hussain, S. A. Alrokayan, S. R. Gottumukkala, R. Digumarti, and V. K. Kutala, "Artificial neural network-based exploration of gene-nutrient interactions in folate and xenobiotic metabolic pathways that modulate susceptibility to breast cancer," *Gene,* vol. 580, no. 2, pp. 159-168, 2016, doi: 10.1016/j.gene.2016.01.023.

[47] Q. Li and M. F. Chan, "Predictive time-series modeling using artificial neural networks for Linac beam symmetry: an empirical study," *Annals of the New York Academy of Sciences,* vol. 1, pp. 84-94, 2016, doi: 10.1111/nyas.13215.

[48] M. Itik, M. U. Salamci, and S. P. Banks, "Optimal control of drug therapy in cancer treatment," *Nonlinear Analysis: Theory, Methods & Applications,* vol. 71, no. 12, pp. 1473-1486, 2009, doi: 10.1016/j.na.2009.01.214.

[49] S. Bunimovich-Mendrazitsky, E. Shochat, and L. Stone, "Mathematical model of BCG immunotherapy in superficial bladder cancer," *Bulletin of Mathematical Biology,* vol. 69, no. 6, pp.1847-1870, 2007, doi: 10.1007/s11538-007-9195-z.

[50] H. Drucker, C. J. Burges, L. Kaufman, A. J. Smola, and V. Vapnik, "Support vector regression machines," *Advances in Neural Information Processing Systems,* vol. 9, pp. 155-161, 1997, doi: 10.1.1.644.3302.

[51] WHA70.12: Cancer prevention and control in the context of an integrated approach, Accessed: Nov. 18, 2020. [Online]. Available: http://apps.who.int/gb/ebwha/pdf_files/WHA70-REC1/A70_2017_REC1-en.pdf
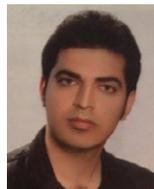
F. Heydarpour

F. Heydarpour received his BSc degree in Mathematics education from Shahid Rajaee University of Kerman'shah, Iran in 2010 and his MSc degree in Numerical Analysis from Shahid Rajaee Teacher Training University of Tehran, Iran in 2012. In 2019, he received his Ph.D. in Applied Mathematics from Yazd University, Iran. He has accepted several international papers in high rank journals. His research interests include numerical optimization, operations research, neural networks, numerical analysis, and optimal control.

E. Abbasi

E. Abbasi received the M.Sc. in computer science from Department of Computer Science of Amirkabir University of Technology, Tehran, Iran, in 2011. In 2016, she earned her PhD from Amirkabir University of Technology. Now, she is an assistant professor with the Department of Computer Science of Yazd University. Her major is data mining, soft computing, intelligent systems and bioinformatics.

M. J. Ebadi

M. J. Ebadi received his BSc degree in Applied Mathematics from Shahid Bahonar University of Kerman, Iran in 2003 and his MSc degree in Applied Mathematics from S&B University, Iran in 2006. In the same year, he joined the Department of Mathematics at Chabahar Maritime University, Iran, as a faculty member. In 2018, he received his Ph.D. in Applied Mathematics from Yazd University, Iran. He is a member of editorial board of two international reputed journals and a reviewer of more than 20 international reputed journals. He has published several international papers in high rank journals. Currently, he is assistant professor in Applied Mathematics at the Department of Mathematics, Chabahar Maritime University, Iran. His research interests include numerical optimization, deep learning, neural networks, numerical analysis, optimal control, fuzzy optimization, EEG signals classification, and image processing.

S. M. Karbassi

Professor S. M. Karbassi obtained his BSc degree in Electrical Engineering (1969) and MSc degree in Engineering Mathematics (1970) from University of Newcastle, England. He began his academic career at Isfahan University as an instructor at the Physics Department. Later, he obtained his PhD degree in Control Engineering from University of Salford, UK (1978). He is now a retired Professor of Applied Mathematics and Control of Yazd University. He has continued his research since his retirement (2007) by supervising research of postgraduate students mostly in different areas of Engineering Mathematics and Control.